PROGRAMSKI JEZICI 2 (A401) – PISMENI DIO ISPITA (08.09.2016.)

1. **(30)** Muzički zapisi mogu da se klasifikuju po žanrovima, s tim što jedan muzički zapis može da ima više žanrova. Svaki muzički zapis ima sljedeće karakteristike: naziv, izvođač, vrijeme trajanja i godina objavljivanja. Muzički žanrovi mogu biti rok, pop i džez. Muzički zapisi mogu da se podijele na instrumentale i zapise sa tekstom. Instrumentale karakteriše YouTube link i oni mogu biti samo džez žanra, dok zapise sa tekstom karakteriše tekst pjesme i ovi zapisi mogu bit rok, pop i džez žanra. Muzički album predstavlja kolekciju muzičkih zapisa takvu da se na jednom albumu mogu pojaviti i instumentali i zapisi sa tekstom. Karaoke četvrt je predstavljena matricom 5x10, gdje se na slučajno odabranim lokacijama postavlja 7 muzičkih albuma. Prije postavljanja na matricu, muzički albumi se kreiraju kroz komandnu liniju. Radi lakšeg testiranja broj muzičkih zapisa na albumu ograničiti na 3. Nakon postavljanja albuma na matricu, kreće simulacija. Karaoke učesnici (proizvoljno odabrani broj) kreću se redovima matrice prelazeći slučajno odabrani broj polja u opsegu od 1 do 3. Kada naiđu na muzički album, prikazuje im se sadržaj albuma vodeći računa o tome da se prikazuju samo zapisi sa tekstom i redni broj zapisa, pa se traži da učesnik odabere koji zapis želi da izvodi. Nakon odabira zapisa, na konzoli se prikazuje tekst zapisa, po principu da se prikazuje riječ po riječ sa pauzom između prikaza riječi onoliko sekundi koliko je prethodna riječ dugačka. Kada se prikaže sav tekst, učesnik nastavlja kretanje dalje kroz matricu po istom principu. Simulacija završava kada svi učesnici dođu do kraja matrice. Na kraju simulacije potrebno je ispisati učesnike sortirane po broju “otpjevanih” zapisa od najvećeg ka najmanjem.

1. **(25)** *HTML parser* je aplikacija koja služi za parsiranje HTML dokumenata koji se nalaze postavljeni na udaljenoj lokaciji zadatoj URL-om. Aplikacija se sastoji od tri servera. Prvi server, *GetHTMLContentAndStoreIt,* na osnovu URL-a koji mu prosljeđuje klijent, sa udaljene lokacije preuzima HTML fajl i smješta ga na fajl sistem u folder sa trenutnim datumom, naziva jednakog nazivu na udaljenoj lokaciji. Za preuzimanje HTML sadržaja sa udaljene lokacije koristiti klasu URL i metodu *openStream*. Drugi server, *GetLinks*, očitava HTML fajl iz foldera sa trenutnim datumom i pronalazi sve linkove koji se nalaze u učitanom sadržaju. Zbog pojednostavljenja implementacije pretpostaviti da prvi i drugi server dijele zajedničko skladište podataka. Potrebno je pronaći samo linkove koji se nalaze na HTML stranici kao vrijednost *href* atributa <a> taga, tj. tekst koji se nalazi nakon href=” pa do sljedećeg zatvorenog navodnika ”, npr. <a href="**http://www.etfbl.net**" target="\_blank">Elektrotehnički fakultet Banjaluka</a>. Očitani linkovi se smještaju u proizvoljno odabranu kolekciju, tako da ne postoje duplikati, pa se kolekcija serijalizuje u fajl *linkovi\_datum.ser*. Serijalizovana kolekcija se prosljeđuje trećem serveru. Treći server učitava serijalizovane linkove, prolazi kroz kolekciju i pronalazi sve linkove koji počinju sa *http*, pa za svaki pronađeni link poziva ponovo prvi server koji preuzima HTML sadržaj sa lokacija predstavljenih linkom i smješta ih u folder *dostupniLinkovi* u folder sa trenutnim datumom. Kada prvi server završi sa proceriranjem dobijenih linkova, aplikacija završava sa radom. Za testiranje aplikacije kao početni URL za prvu nit iskoristiti <http://www.s-cube.rs.ba/sr/index.html>.
2. **(15)** Proširiti klasu *File* sa dodatnim atributom *owner* i metodom za provjeru da li fajl pripada zadatom vlasniku. Vlasnik fajla se podešava prilikom kreiranja fajla. Za očitavanje vlasnika fajla sa sistema koristiti klasu *System* i *property* **user.name**. Za novonastalu klasu potrebno je implementirati komparator koji vrši poređenje fajlova po nazivu. Napraviti RMI aplikaciju koja omogućava kreiranje proizvoljnog broja novonastalih fajlova tako što se kao argument komandne linije prosljeđuje broj fajlova koje je potrebno kreirati, a po pokretanju se zadati broj puta od korisnika traži unos naziva i sadržaja fajla, te se fajlovi prosljeđuju serveru u proizvoljno odabranoj kolekciji. Po završetku kreiranja fajlova, na klijentskoj strani se prikazuju fajlovi sortirani po nazivu. Sortirani fajlovi se dobijaju sa RMI servera.